

PAT-NO: JP408095810A
DOCUMENT-IDENTIFIER: JP 08095810 A
TITLE: BATCH JOB EXECUTION SYSTEM
PUBN-DATE: April 12, 1996

INVENTOR-INFORMATION:
NAME
MIYAZAWA, HIROSHI

ASSIGNEE-INFORMATION:
NAME COUNTRY
NEC CORP N/A

APPL-NO: JP06258713
APPL-DATE: September 28, 1994

INT-CL (IPC): G06F009/46, G06F015/00

ABSTRACT:

PURPOSE: To process plural job steps simultaneously in parallel by using a buffer for data transmission on a main storage for data transmission between job steps constituting a batch job instead of a temporary file.

CONSTITUTION: Instead of the common temporary file that a job step 7-1 uses for output and a job step 7-2 uses for input respectively, the FIFO buffer 5-1 for data transmission is prepared on the main storage. A step parallel execution management part 4 allocates different processes 6-1 and 6-2 to the job steps 7-1 and 7-2 and places them into parallel operation. A transmitted

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平8-95810

(43)公開日 平成8年(1996)4月12日

(51)Int. CL ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/46	3 4 0 B	7737-5B		
15/00	3 1 0 M	9364-5L		

審査請求 有 請求項の数3 F D (全 12 頁)

(21)出願番号 特願平6-258713

(22)出願日 平成6年(1994)9月28日

(71)出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72)発明者 宮澤 浩

東京都港区芝五丁目7番1号 日本電気株式会社内

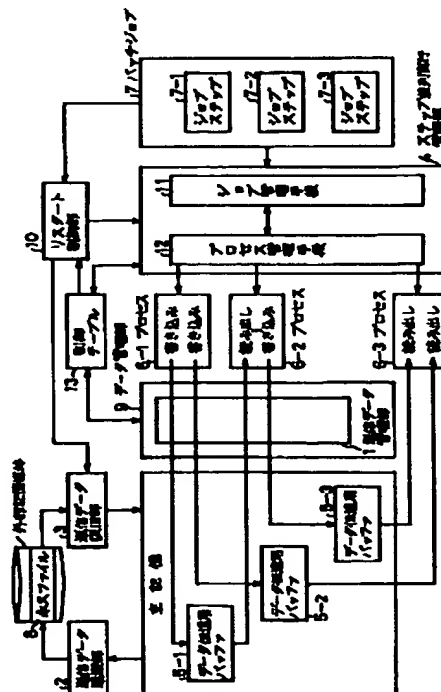
(74)代理人 弁理士 境 廣巳

(54)【発明の名称】 バッチジョブ実行方式

(57)【要約】

【目的】 バッチジョブを構成するジョブステップ間のデータ伝達に一時ファイルの代わりに主記憶上のデータ伝達用バッファを使用させ、複数ジョブステップを同時並列に処理する。

【構成】 ジョブステップ7-1が出力に、ジョブステップ7-2が入力にそれぞれ使用する共通の一時ファイルの代わりに主記憶上に先入り先出し形式のデータ伝達用バッファ5-1を用意する。ステップ並列実行管理部4はジョブステップ7-1,7-2に別々のプロセス6-1,6-2を割り当てて並行して動作させる。送信データ管理部1は、プロセス6-1から一時ファイルに対する書き込み要求があったときに出力データをデータ伝達用バッファ5-1に順次にキューイングし、プロセス6-2から一時ファイルに対する読み込み要求があったときにデータ伝達用バッファ5-1にキューイングされたデータを順次に取り出してプロセス6-2に伝達する。



1

【特許請求の範囲】

【請求項1】 複数のデータを処理して個々のデータ毎の処理済データを順次に一時ファイルに出力する第1のジョブステップと、前記一時ファイルから個々の処理済データを順次に入力して処理する第2のジョブステップとを含むバッチジョブの実行方式において、前記第1のジョブステップと前記第2のジョブステップとにそれぞれ別々のプロセスを割り当てて並行して動作させるステップ並列実行管理部と、前記一時ファイルの代わりに主記憶上に設けられた先入れ先出し形式のデータ伝達用バッファと、前記第1のジョブステップに割り当てられたプロセスから前記一時ファイルに対する書き込み要求があったときに出力データを前記データ伝達用バッファに順次にキューイングし、前記第2のジョブステップに割り当てられたプロセスから前記一時ファイルに対する読み込み要求があったときに前記データ伝達用バッファにキューイングされたデータを順次に取り出して前記第2のジョブステップに割り当てられたプロセスに伝達する送信データ管理部とを備えることを特徴とするバッチジョブ実行方式。

【請求項2】 前記データ伝達用バッファの内容の履歴を前記プロセスの動作と並行して永久ファイルに退避する送信データ退避部を備えることを特徴とする請求項1記載のバッチジョブ実行方式。

【請求項3】 障害発生等の原因によりジョブの実行が停止した後、途中終了したジョブステップの先頭からジョブを再実行する際、途中終了したジョブステップが入力に使用していた一時ファイルであって正常終了したジョブステップが出力に使用していた一時ファイルに対応するデータ伝達用バッファの内容の履歴を前記永久ファイルから読み出し、前記正常終了したジョブステップが出力した全てのデータを格納したデータ伝達用バッファを主記憶上に復元する送信データ復旧部を備えることを特徴とする請求項2記載のバッチジョブ実行方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は計算機のバッチジョブ実行方式に関し、特にジョブステップ間で一時ファイルを介してデータの伝達を行う複数のジョブステップを、その一時ファイルの代わりに主記憶上のデータ伝達用バッファを使用して同時に並行して動作させるようにしたバッチジョブ実行方式に関する。

【0002】

【従来の技術】従来、複数のジョブステップから構成されるバッチジョブの実行は、それを構成するジョブステップを記述順に順次一つずつ実行することで行われている。

【0003】また、複数のジョブステップから構成されるバッチジョブの実行において、先行するジョブステッ

2

プで生成されたデータを後続のジョブステップで処理する場合、一時ファイルを使用してジョブステップ間でデータの伝達を行っている。一時ファイルとしては、磁気ディスク媒体上のファイル以外に、その入出力速度を向上するために電子ディスク媒体上のファイルが使用される場合がある。

【0004】

【発明が解決しようとする課題】ところで、ジョブの実行速度を速める手法の一つに、オンライントランザクションを処理するリアルタイムジョブにおいて採用されているマルチタスキング手法がある。即ち、各トランザクションの処理を別々のプロセスで並行して処理する手法である。しかしながら、このマルチタスキング手法はバッチジョブに対しては余り利用されていない。その理由の一つは、本来バッチジョブは逐次性の強い処理を目的としていたという事情があったからである。

【0005】しかしながら、例えばPという計算処理とQという計算処理とがあり、それらの計算が互いに独立に行える場合、Pという計算処理とQという計算処理とを行う1つのジョブステップを1つのプロセスで実行するよりも、Pという計算処理を行うプロセスとQという計算処理を行うプロセスとの2つのプロセスで並行処理の方が計算結果が早く得られる。つまり、バッチジョブにおいてもマルチタスキング手法が有効な場合がある。

【0006】同様に、上記のPという計算処理を行うジョブステップと、Qという計算処理を行うジョブステップとがある場合、ジョブステップの実行を逐次的に実行するのではなく、双方のジョブステップに別々のプロセスを割り当てて並列して実行させることも考えられる。

【0007】しかしながら、上記の何れの場合も各ジョブステップでの計算処理が互いに独立に行えるので問題はないが、先行するジョブステップで生成されたデータを後続のジョブステップで処理するようなバッチジョブの場合、従来はデータ伝達用に一時ファイルを使用していた為に実質的にジョブステップは並行処理されない。即ち、一時ファイルは、ファイルの構造上の問題により、全てのデータが書き出され、そのファイルが完成してからでないと、データの読み込みができないため、先行するジョブステップが全てのデータを一時ファイルに書き出した後に、後続するジョブステップがその読み込みを開始することになり、ジョブステップの実行が逐次的に成らざるを得ないからである。

【0008】そこで本発明の目的は、一時ファイルの代わりに主記憶上に個々のデータを先入れ先出し形式で格納するデータ伝達用のバッファを設け、先行するジョブステップと後続のジョブステップとが非同期に個々のデータ単位にデータ伝達し得るようにすることにより、複数のジョブステップを実質的に同時に並行して動作させることを可能として、バッチジョブの実行時間を短縮し

3

得るようにしたバッチジョブ実行方式を提供することにある。

【0009】

【課題を解決するための手段】本発明は上記の目的を達成するために、複数のデータを処理して個々のデータ毎の処理済データを順次に一時ファイルに出力する第1のジョブステップと、前記一時ファイルから個々の処理済データを順次に入力して処理する第2のジョブステップとを含むバッチジョブの実行方式において、前記第1のジョブステップと前記第2のジョブステップとにそれぞれ別々のプロセスを割り当てて並行して動作させるステップ並列実行管理部と、前記一時ファイルの代わりに主記憶上に設けられた先入れ先出し形式のデータ伝達用バッファと、前記第1のジョブステップに割り当てられたプロセスから前記一時ファイルに対する書き込み要求があったときに出力データを前記データ伝達用バッファに順次にキューイングし、前記第2のジョブステップに割り当てられたプロセスから前記一時ファイルに対する読み込み要求があったときに前記データ伝達用バッファにキューイングされたデータを順次に取り出して前記第2のジョブステップに割り当てられたプロセスに伝達する送信データ管理部とを備えている。

【0010】また、障害発生時等に主記憶上のデータ伝達用バッファの内容を復元できるようにするために、データ伝達用バッファの内容の履歴を前記プロセスの動作と並行して永久ファイルに退避する送信データ退避部を備えている。そして、障害発生等の原因によりジョブの実行が停止した後、途中終了したジョブステップの先頭からジョブを再実行する際、送信データ復旧部が、途中終了したジョブステップが入力に使用していた一時ファイルであって正常終了したジョブステップが出力に使用していた一時ファイルに対応するデータ伝達用バッファの内容の履歴を前記永久ファイルから読み出し、前記正常終了したジョブステップが出力した全てのデータを格納したデータ伝達用バッファを主記憶上に復元するようにしている。

【0011】

【作用】本発明のバッチジョブ実行方式においては、一時ファイルの代わりに主記憶上に先入り先出し形式のデータ伝達用バッファが設けられ、ステップ並列実行管理部が、複数のデータを処理して個々のデータ毎の処理済データを順次に出力する第1のジョブステップと、この第1のジョブステップから出力された個々の処理済データを順次に処理する第2のジョブステップとにそれぞれ別々のプロセスを割り当てて並行して動作させ、送信データ管理部が、第1のジョブステップに割り当てられたプロセスから前記一時ファイルに対する書き込み要求があったときに出力データを前記データ伝達用バッファに順次にキューイングし、前記第2のジョブステップに割り当てられたプロセスから前記一時ファイルに対する

4

読み込み要求があったときに前記データ伝達用バッファにキューイングされたデータを順次に取り出して前記第2のジョブステップに割り当てられたプロセスに伝達する。

【0012】

【実施例】次に本発明の実施例について図面を参照して詳細に説明する。

【0013】図1は本発明のバッチジョブ実行方式の一実施例のブロック図であり、1は送信データ管理部、2は送信データ退避部、3は送信データ復旧部、4はジョブ管理手段11およびプロセス管理手段12を含むステップ並列実行管理部、5-1~5-3は主記憶上のデータ伝達用バッファ、6-1~6-3はプロセス、7はバッチジョブ、7-1~7-3はバッチジョブ7を構成するジョブステップ、8は磁気ディスク媒体等の外部記憶媒体上の永久ファイル、9はデータ管理部、10はリスタート制御部、13は不揮発性の制御テーブルである。このような構成要素は、マルチプロセッサ計算機の如き複数のプロセスを同時に並行して実行することができる複数プロセス処理機能搭載計算機上に搭載される。

【0014】図2はバッチジョブ7のジョブステップ7-1~7-3で行う処理例の説明図である。同図に示すように、ジョブステップ7-1は、複数のデータA1~ANを順次に処理して、個々のデータ毎に処理済データB1~BNおよび処理済データC1~CNを生成し、前者のデータB1~BNは一時ファイルXに格納し、後者のデータC1~CNは一時ファイルYに格納する。ジョブステップ7-2は、一時ファイルXからデータB1~BNを順次に入力して処理し、個々のデータ毎に処理済データD1~DNを生成して一時ファイルZに格納する。ジョブステップ7-3は一時ファイルYおよび一時ファイルZから処理済データC1~CN、D1~DNを順次に入力して処理する。本実施例では、このような一時ファイルX、Y、Zを介してデータ授受し合う3つのジョブステップ7-1~7-3を並列に動作させようとするものである。

【0015】図3は各ジョブステップ7-1~7-3のJCL記述例を示す。ジョブステップ7-1は一時ファイルX、Yを使用するので、そのJCL記述には一時ファイルX、Yに関するアサイン文\$ASSIGNがあり、ジョブステップ7-2は一時ファイルX、Zを使用するので、そのJCL記述には一時ファイルX、Zに関するアサイン文\$ASSIGNがある。また、ジョブステップ7-3は一時ファイルY、Zを使用するので、そのJCL記述には一時ファイルY、Zに関するアサイン文\$ASSIGNがある。更に、ジョブステップ7-1のJCL記述中には、当該ジョブステップ7-1とその直後のジョブステップ7-2とを並列に動作させることを指示する文\$PARALLELがあり、ジョブステップ7-2のJCL記述中には、当該ジョブステップ7-

5

2とその直後のジョブステップ7-3とを並列に動作させることを指示する文\$PARALLELがある。なお、ジョブステップ7-3のJCL記述中には文\$PARALLELはない。

【0016】図4はジョブステップ7-1の処理の一例を示すフローチャートである。同図に示すように、先ず、一時ファイルX、Yをオープンし(S1)、次にSYSINファイル等から図2に示したデータA1~ANを1データずつ入力し(S2)、所定のデータ処理を行う(S4)。そして、処理結果のデータを一時ファイルXに書き出し(S5)、再びステップS2で入力したデータに対しステップS4とは別のデータ処理を行って(S6)、その結果のデータを一時ファイルYに書き出す(S7)。以上を最後のデータANまで実行すると(S3でYES)、一時ファイルX、Yをクローズし(S8)、処理を終了する。

【0017】図5はジョブステップ7-2の処理の一例を示すフローチャートである。先ず、一時ファイルX、Zをオープンし(S11)、次に一時ファイルXからデータB1~BNを1つずつ入力し(S12)、所定のデータ処理を行う(S14)。次に、その結果のデータを一時ファイルZに書き込む(S15)。以上を一時ファイルXの最後のデータBNまで実行すると(S13でYES)、一時ファイルX、Zをクローズし(S16)、処理を終了する。

【0018】図6はジョブステップ7-3の処理の一例を示すフローチャートである。先ず、一時ファイルY、Zをオープンし(S21)、次に一時ファイルY、ZからデータC1~CN、D1~DNを1つずつ入力し(S22、S24)、所定のデータ処理を行う(S25)。そして、その結果のデータを例えばSYSOUTファイル等に出力する(S26)。以上を一時ファイルYの最後のデータCNまで実行すると(S23でYES)、一時ファイルY、Zをクローズし(S27)、処理を終了する。

【0019】図7はステップ並列実行管理部4のジョブ管理手段11の処理例を示すフローチャートである。同図に示すようにジョブ管理手段11は、バッチジョブ7のジョブステップ記述(JCL記述)をその先頭より順に入力し(S31)、入力したジョブステップ記述を解析して前処理を実行する(S33)。この前処理では、当該ジョブステップに必要な資源の確保等の処理が行われる。このとき本実施例では、一時ファイルのアサイン文\$ASSIGNが現れたとき、実際に一時ファイルを確保するのではなく、それに代わって主記憶上にデータ伝達用バッファを確保する。また、どの一部ファイルに対しどのデータ伝達用バッファを割り当てたかを示す情報を作成し、制御テーブル13に設定する。次に、当該ジョブステップを実行するためのプロセスを割り当て、プロセス管理手段12に対しそのプロセスの実行を指示

6

する(S34)。次に、当該ジョブステップ記述中に\$PARALLEL文が存在するか否かを調べることであり、後続ジョブステップの並列実行が指示されているか否かを調べ(S35)、指示されていればステップS31に戻って次のジョブステップ記述について上記と同様な処理を繰り返す。

【0020】他方、並列実行が指示されていないジョブステップ記述を処理し終えた場合(S35でNO)、上記実行を開始させたジョブステップ対応のプロセスの実行終了を待ち合わせ(S36)、何れかのプロセスが終了すると、その終了したプロセスに対応するジョブステップの後処理を行う(S37)。この後処理では、そのジョブステップが使用していた資源の解放等を行う。但し、他のジョブステップと共用される資源であるデータ伝達用バッファの解放はジョブの終了時までには解放されない。次に、他のジョブステップ対応のプロセスのうち実行中のプロセスが残っているか否かを判別し(S38)、残っていればその終了を待ち合わせるためにステップS36へ進む。残っていなければ、ステップS31に戻って、後続のジョブステップ記述を処理する。

【0021】図1に示されるデータ伝達用バッファ5-1~5-3は上記のジョブ管理手段11によって一時ファイルの代わりに主記憶上に確保されたもので、データ伝達用バッファ5-1は一時ファイルXに、データ伝達用バッファ5-2は一時ファイルYに、データ伝達用バッファ5-3は一時ファイルZに対応している。このような関係は、図8に示すように制御テーブル13に保持される。なお、制御テーブル13にはステータスを登録する欄があり、この欄に送信データ管理部1によってどの一時ファイルがどのプロセスによって使用中であるか等のステータスが設定される。

【0022】図9はデータ伝達用バッファ5-1~5-3の構成例を示すブロック図であり、各々のデータ伝達用バッファ5-1~5-3はキュー511、521、531を有している。データ伝達用バッファ5-1へのデータ書き込み時、書き込まれたデータの主記憶上のアドレスを示すポインタがキュー511にキューイングされ、データの読み出し時には、キュー511からポインタがデキューイングされ、そのポインタが指し示すデータが読み出し要求元に伝達される。他のデータ伝達用バッファ5-2、5-3も同様の構成を有している。

【0023】図10~図13は送信データ管理部1の処理例を示すフローチャートである。図10に示すように、送信データ管理部1は実行中のプロセスから一時ファイルに対するオープン要求があると、制御テーブル13中のオープン要求された一時ファイルに対応するステータス欄に、その要求元が使用中である旨を登録する(S41)。また図11に示すように、実行中のプロセスから一時ファイルに対するクローズ要求があると、制御テーブル13中のクローズ要求された一時ファイルに

対応するステータス欄に、その要求元が使用を終えた旨を記述する(S51)。

【0024】更に図12に示すように、送信データ管理部1は実行中のプロセスから一時ファイルに対するデータ出力要求があると、制御テーブル13を参照して出力要求された一時ファイルに対応するデータ伝達用バッファを認識し(S61)、出力要求されたデータを1オブジェクトとして上記のデータ伝達用バッファに格納する(S62)。また、図1の送信データ退避部2に、格納先のデータ伝達用バッファ名と格納したデータとを含む履歴データを通知する(S63)。

【0025】また図13に示すように、送信データ管理部1は実行中のプロセスから一時ファイルに対するデータ入力要求があると、制御テーブル13を参照して入力要求された一時ファイルに対応するデータ伝達用バッファを認識し(S71)、そのデータ伝達用バッファから1つのデータを取り出して要求元に伝達する(S73)。このとき、データ伝達用バッファに1つもデータが格納されていないときは(S72でNO)、制御テーブル13中のそのデータ伝達用バッファに対応するステータス欄の記述に基づき、入力要求元プロセス以外のプロセス(即ち、データ出力元のプロセス)がそのデータ伝達用バッファの使用を終えたか否かを調べ(S74)、使用を終えていなければ所定時間だけ待ち合わせて(S75)、再び処理S72から再試行する。他方、他のプロセスが使用を終えていれば、全てのデータを伝達し終えたことになるので、入力要求元にデータ終了を通知する(S76)。

【0026】図14は送信データ退避部2の処理例を示すフローチャートである。同図に示すように送信データ退避部2は、送信データ管理部1から履歴データが通知されると、その履歴データを永久ファイル8に書き込む(S81)。従って、永久ファイル8には図15に示すようにデータ伝達用バッファ名およびデータを含む履歴データが順次格納されることになる。

【0027】図16はリスタート制御部10の処理例を示すフローチャートである。同図に示すようにリスタート制御部10は、途中終了したジョブステップの入力用であって、それを出力用に使用しているジョブステップが正常終了している一時ファイルを認識し、それに対応するデータ伝達用バッファを主記憶に確保し(S91)、そのデータ伝達用バッファの格納データを復元するよう送信データ復旧部3に要求する(S92)。そして、途中終了したジョブステップの先頭からジョブを再実行するようステップ並列実行管理部4に指示を出す(S93)。

【0028】図17は送信データ復旧部3の処理例を示すフローチャートである。同図に示すように送信データ復旧部3は、永久ファイル8から復元するデータ伝達用バッファ名を含む履歴データを全て抽出し(S10

1)、抽出した履歴データ中のデータを、復元するデータ伝達用バッファに格納する(S102)。

【0029】以下、上述のように構成された本実施例の動作を説明する。

【0030】図3に示したようなジョブ記述が与えられてジョブの実行が指示されると、ステップ並列実行管理部4のジョブ管理手段11はジョブ記述の先頭文から順に解析を開始する。そして、先頭のジョブステップ記述を入力すると(図7のS31)、ジョブステップ実行前処理を行う(S33)。今、先頭のジョブステップがジョブステップ7-1であった場合、その記述中には一時ファイルX、Yのアサイン文が存在するため、一時ファイルXに代えてデータ伝達用バッファ5-1が主記憶上に確保され、一時ファイルYに代えてデータ伝達用バッファ5-2が主記憶上に確保される。また、一時ファイルXがデータ伝達用バッファ5-1に対応し、一時ファイルYがデータ伝達用バッファ5-2に対応していることが図8に示すように制御テーブル13に登録される。ジョブステップ実行の前処理が終了すると、次にそのジョブステップ7-1に1つのプロセスが割り当てられ、プロセス管理手段12に対しプロセスの実行が指示される(S34)。今、ジョブステップ7-1に対してプロセス6-1が割り当てられたとすると、以後、プロセス管理手段12の管理の下で、ジョブステップ7-1がプロセス6-1として実行されることになる。

【0031】ジョブ管理手段11は、次にジョブステップ7-1の記述中に\$PARALLEL文が存在するか否かを調べる(S35)。図3の場合、存在するので、ジョブ管理手段11は次のジョブステップ7-2のジョブステップ記述を入力し(S31)、それを解析して前処理を行う(S33)。ここでは一時ファイルX、Zが宣言されているので、それに代わるデータ伝達用バッファを主記憶上に確保する。但し、一時ファイルXは既にデータ伝達用バッファ5-1として確保済なので、一時ファイルZに代えて使用するデータ伝達用バッファ5-3のみを主記憶上に確保する。また、一時ファイルZがデータ伝達用バッファ5-3に対応することを図8に示すように制御テーブル13に設定する。そして、ジョブステップ7-2に1つのプロセスを割り当て、プロセス管理手段12の管理の下に実行させる(S34)。従って、ジョブステップ7-2にプロセス6-2が割り当てられたとすると、ジョブステップ7-2がプロセス6-2として実行されることになる。

【0032】ジョブ管理手段11は、次にジョブステップ7-2の記述中に\$PARALLEL文が存在するので、次のジョブステップ7-3のジョブステップ記述を入力し(S31)、それを解析して前処理を行う(S33)。ここでは一時ファイルY、Zが宣言されているが、それらに対応するデータ伝達用バッファ5-2、5-3は既に確保済なので、新たなデータ伝達用バッファ

の確保は行わない。そして、ジョブステップ7-3に1つのプロセス、即ちプロセス6-3を割り当て、プロセス管理手段12の管理の下に実行させる(S34)。

【0033】次にジョブ管理手段11は、ジョブステップ7-3の記述中に\$PARALLEL文が存在するか否か調べるが(S35)、図3の例では存在しないので、プロセス6-1~6-3の実行終了を待ち合わせる(S36)。

【0034】以上の結果、図4に示すような処理を行うジョブステップ7-1と、図5に示すような処理を行うジョブステップ7-2と、図6に示すような処理を行うジョブステップ7-3との3つのジョブステップが同時に並行して実行されることになる。

【0035】さて、ジョブステップ7-1の実行体であるプロセス6-1の図4の処理において、一時ファイルX、Yに対するオープン要求があると(S1)、それが、ファイル処理全般についてのOS機能部であるデータ管理部9に伝達される。データ管理部9はオープン要求対象のファイル名が制御テーブル13に登録されているか否かを調べ、登録されている場合には制御を送信データ管理部1に渡す。登録されていないファイルについては従来と同様に処理される。このような切り分けは、クローズ要求時およびデータ入出力要求時にも行われる。従って、一時ファイルX、Yに対するオープン要求が出された場合、制御が送信データ管理部1に渡り、送信データ管理部1は図10に示す処理を実行することにより、制御テーブル13中の一時ファイルX、Yに対応するステータス欄に、プロセス6-1が使用中である旨を設定する。

【0036】プロセス6-1の処理が進み、図2に示した先頭の入力データA1の処理済結果であるデータB1を一時ファイルXへ出力する要求を出すと、制御が送信データ管理部1に渡り、送信データ管理部1は、図12に示すように、制御テーブル13から一時ファイルXに対応するデータ伝達用バッファ5-1を認識し(S61)、出力要求されたデータB1をデータ伝達用バッファ5-1にキューイングする(S62)。即ち、図9に示すように、データB1を主記憶上に書き込むと共に、そのアドレスを示すポインタをキュー511にキューイングする。そして、格納先のデータ伝達用バッファ名と格納したデータB1とを含む履歴データを送信データ退避部2に通知する。送信データ退避部2は通知された履歴データを図15に示すように永久ファイル8に格納する(図14のS81)。

【0037】またプロセス6-1がデータA1を処理して得たデータC1を一時ファイルYへ出力する要求を出すと、送信データ管理部1は、制御テーブル13から一時ファイルYに対応するデータ伝達用バッファ5-2を認識し、図9に示すようにデータC1を主記憶上に書き込むと共に、そのアドレスを示すポインタをキュー52

1にキューイングする。そして、格納先のデータ伝達用バッファ名と格納したデータC1とを含む履歴データを送信データ退避部2に通知し、送信データ退避部2はこの履歴データを永久ファイル8に格納する。

【0038】他方、プロセス6-1と並行して動作しているプロセス6-2において、一時ファイルX、Zに対するオープン要求があると(図5のS11)、送信データ管理部1は制御テーブル13の一時ファイルX、Zに対応するステータス欄に、プロセス6-2が使用中である旨を設定する。また、プロセス6-1の処理が進み、一時ファイルXからのデータ入力要求があると(S12)、送信データ管理部1は、図13に示すように、制御テーブル13を参照して一時ファイルXに対応するデータ伝達用バッファ5-1を認識し(S71)、そこにキューイングされているデータを1つ取り出してプロセス6-2に伝達する(S73)。即ち、図9のキュー511からポインタを1つデキューイングし、それが指示するデータB1をプロセス6-2に伝達する。ここで、プロセス6-1が最初のデータB1を出力する前にプロセス6-2が入力を要求した場合、データ伝達用バッファ5-1には1つもデータが存在しないため、図13のS72でNOと判定される。このときは、制御テーブル13のデータ伝達用バッファ5-1に対応するステータス欄が調べられ、要求元であるプロセス6-2以外のプロセス6-1が未だ使用中であることが確認され、所定時間の待ち合わせ後にプロセス6-2からの入力要求が再処理される。この結果、プロセス6-1からデータB1が出力されると、それがプロセス6-2に伝達されることになる。

【0039】またプロセス6-2がデータB1を処理して得たデータD1を一時ファイルZに出力する要求を出すと、送信データ管理部1は一時ファイルZに対応するデータ伝達用バッファ5-3にデータD1を格納する。また、この格納されたデータD1を含む履歴データが送信データ退避部2によって永久ファイル8に格納される。

【0040】更にプロセス6-1、6-2と並行して動作しているプロセス6-3から一時ファイルY、Zのオープン要求があった場合、一時ファイルY、Zからのデータ入力要求があった場合も、前述と同様の処理が行われ、データ伝達用バッファ5-2、5-3から取り出されたデータがプロセス6-3に伝達される。

【0041】さて、プロセス6-1が図2に示す最後のデータANまで処理し終えたことから、一時ファイルX、Yのクローズ要求を出すと、送信データ管理部1は、制御テーブル13中の一時ファイルX、Yに対応するステータス欄に、プロセス6-1が使用を終えた旨の記述を登録する。

【0042】またプロセス6-2が図2に示す最後のデータBNを処理し終え、一時ファイルXからの次のデー

11

タの入力要求を行うと、送信データ管理部1は一時ファイルXに対応するデータ伝達用バッファ5-1からデータを取り出そうとするが、最後のデータBNまで伝達し終わったので、最早取り出すべきデータは存在しない。このため、送信データ管理部1は図13のS73からS74へ進み、制御テーブル13中のデータ伝達用バッファ5-1に対応するステータス欄を調べる。この時点では、プロセス6-1が使用を終えた旨の記述が登録されているため、送信データ管理部1は、全てのデータを伝達し終えた為にデータ伝達用バッファ5-1が空になっ

【0043】プロセス6-3が最後の入力データCNを処理し終えた後の動作も、前述したプロセス6-2の場合と同じである。

【0044】以上のようにして、データ伝達用バッファ5-1~5-3を使用して非同期にデータ伝達し合いながら、3つのプロセス6-1~6-3、つまり3つの連続するジョブステップ7-1~7-3が並行して実行される。

【0045】バッチジョブ7はその最後のジョブステップまで正常に実行し終えると、終了する。但し、何れかのジョブステップの実行途中で障害が発生し、処理の続行が不可能になると、ジョブの実行は異常終了する。このようなときは、途中終了したジョブステップの先頭からジョブを再実行する必要がある。本実施例では、一時ファイルの代わりに主記憶上のデータ伝達用バッファを使用しているの

【0046】リスタート制御部10は、リスタート制御の一環として、異常終了したジョブの情報を調べて途中終了したジョブステップを認識し、この途中終了したジョブステップが入力に使用している一時ファイルであって、それを出力用に使用するジョブステップが正常終了している一時ファイルに対応するデータ伝達用バッファを図8の制御テーブル13から調べる。そして、図16に示すように、そのデータ伝達用バッファを主記憶上に確保し(S91)、そのデータ伝達用バッファ名を送信データ復旧部3に通知して復元を要求する(S92)。また、制御テーブル13中の当該確保したデータ伝達用バッファに関連する部分のみを残して他は消去し、新たな制御テーブル13を作成する。送信データ復旧部3は、図17に示すように、永久ファイル8から、通知さ

12

れたデータ伝達用バッファ名を含む履歴データを全て抽出し(S101)、この抽出した履歴データ中のデータを、復元要求された主記憶上のデータ伝達用バッファに格納する(S102)。その後、リスタート制御部10は、途中終了したジョブステップの先頭からジョブを再実行するようステップ並列実行管理部4に指示する(S93)。

【0047】従って、例えばジョブステップ7-1~7-3が並行して実行されている状態で、ジョブステップ7-1のみが正常に終了し、ジョブステップ7-2、7-3が途中終了した場合、途中終了したジョブステップ7-2、7-3が入力に使用している一時ファイルであって、それを出力用に使用するジョブステップが正常終了している一時ファイルは、一時ファイルXと一時ファイルYなので、それらに対応するデータ伝達用バッファ5-1、5-2が主記憶上に確保され、送信データ復旧部3により、永久ファイル8に格納されているデータ伝達用バッファ5-1の全ての履歴データが抽出されてその中のデータがデータ伝達用バッファ5-1に格納され、また永久ファイル8に格納されているデータ伝達用バッファ5-1の全ての履歴データが抽出されてその中のデータがデータ伝達用バッファ5-2に格納される。そして、ジョブステップ7-2からジョブが再実行される。

【0048】以上本発明の実施例について説明したが、本発明は以上の実施例にのみ限定されずその他各種の付加変更が可能である。例えば、後続のジョブステップを並列に実行すべき指示を直前のジョブステップ記述中に1つのジョブステップ文(\$PARALLEL)として挿入したが、ジョブステップ記述間に挿入するようにしても良い。また、後続のジョブステップ記述の先頭の\$STEP文中に直前のジョブステップとの同時実行を指示するパラメータを挿入するようにしても良い。

【0049】

【発明の効果】以上説明した本発明のバッチジョブ実行方式によれば、以下のような効果を得ることができる。

【0050】ジョブステップ間のデータ伝達用に、個々のデータを独立してキューイング、デキューイングできる主記憶上のデータ伝達用バッファを使用したので、一時ファイルを使用した場合のように先行するジョブステップが全てのデータを出力した後に後続のジョブステップがデータの読み出しを開始しなければならないような制約がなくなり、先行するジョブステップが個々の処理済データを出力する毎に、その処理済データを後続のジョブステップが読み込むことができ、複数のジョブステップを実質的に同時に並行して動作させることが可能となる。これにより、バッチジョブ全体の実行時間を短縮することができる。

【0051】一時ファイルの代わりに主記憶上のデータ伝達用バッファを使用すると、障害発生時等のデータ消

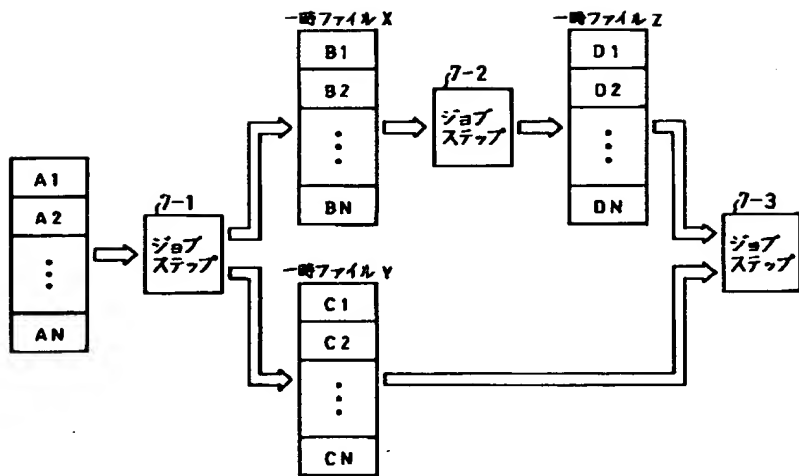
14

1.3...制御テーブル

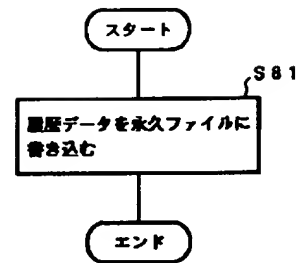
【図11】送信データ管理部のファイルクローズ要求時

[illegible]

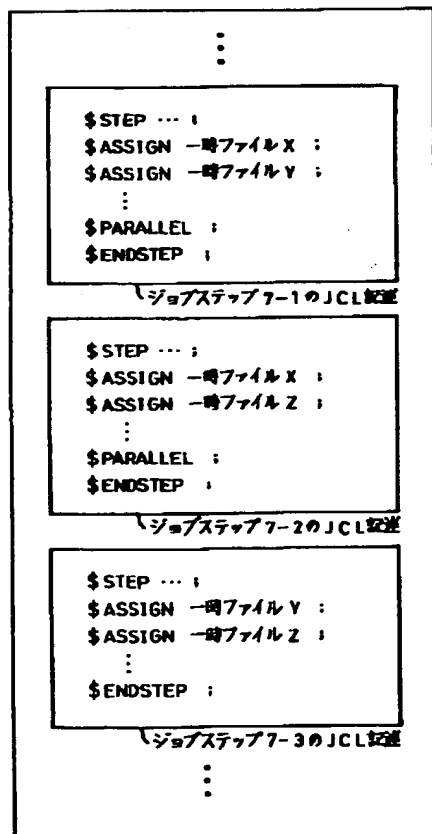
【図2】



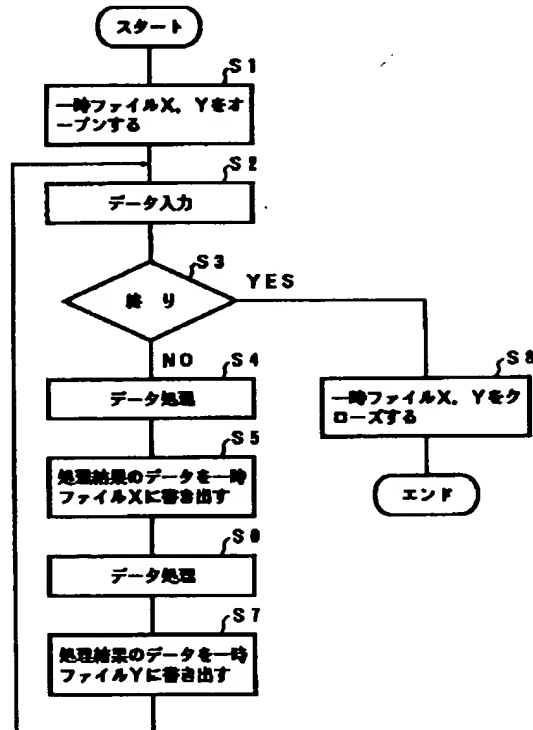
【図14】



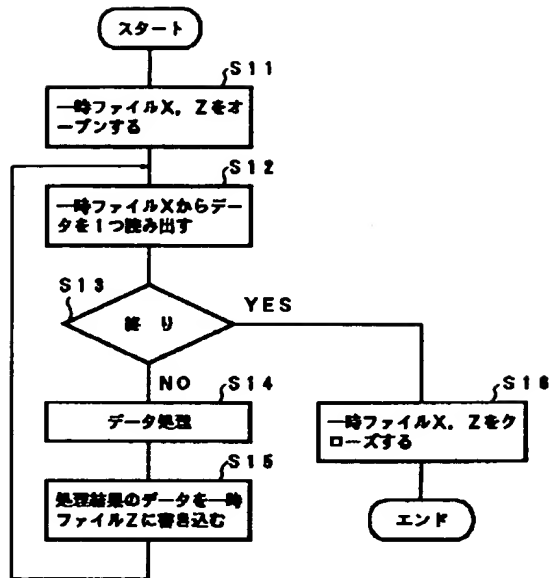
【図3】



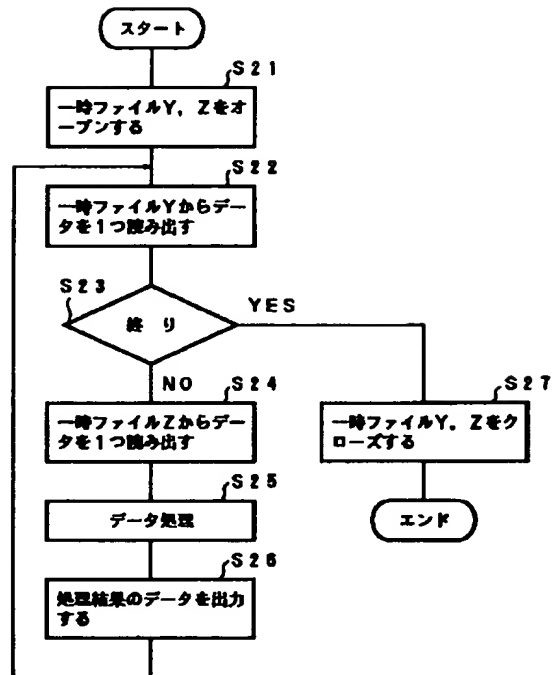
【図4】



【図5】



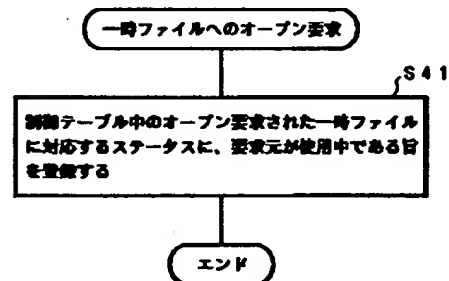
【図6】



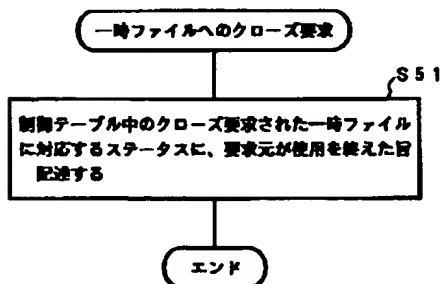
【図8】

一時ファイル名	データ伝達用バッファ名	ステータス
一時ファイルX	データ伝達用バッファ名5-1	
一時ファイルY	データ伝達用バッファ名5-2	
一時ファイルZ	データ伝達用バッファ名5-3	

【図10】



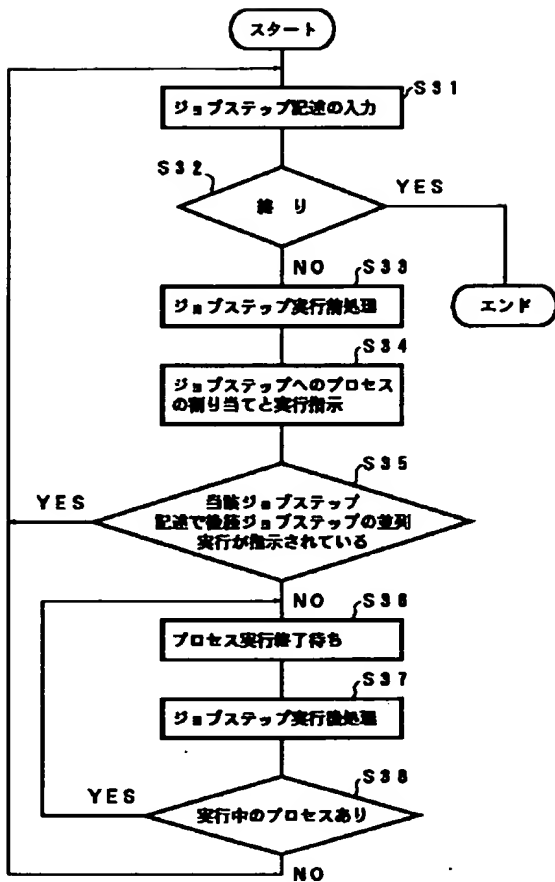
【図11】



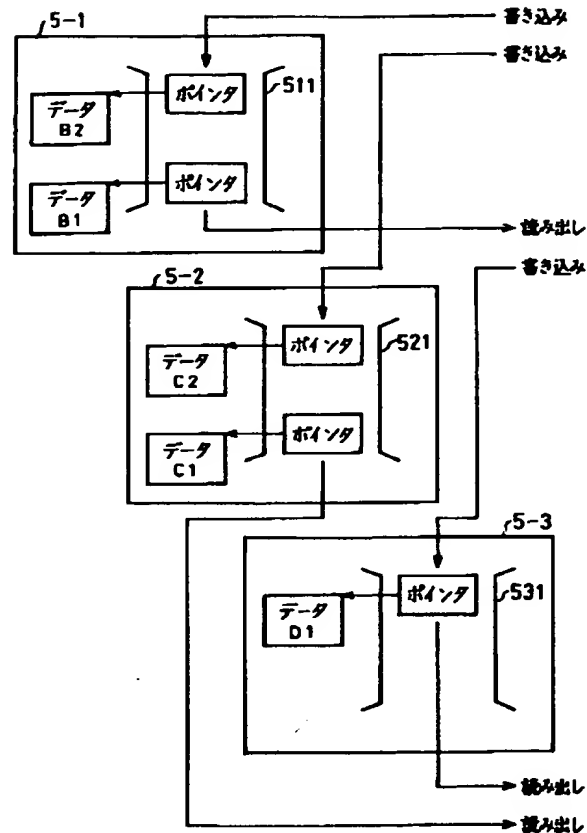
【図15】

履歴データ	
データ伝達用バッファ名	データ
・	・
・	・
・	・
・	・

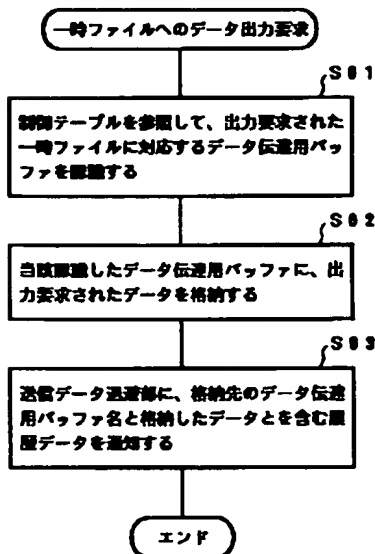
【図7】



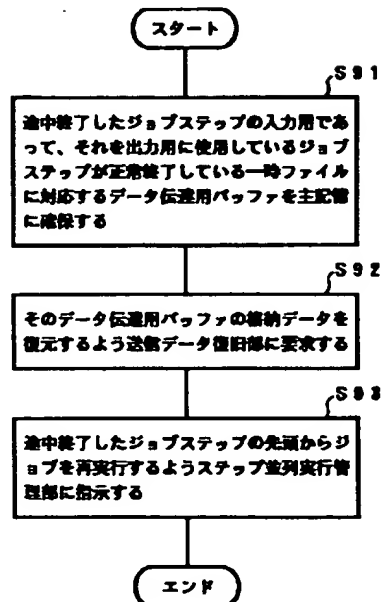
【図9】



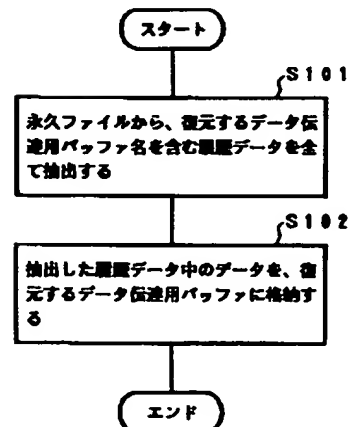
【図12】



【図16】



【図17】



【図13】

